Journal of Siberian Federal University. Mathematics & Physics 2009, 2(1), 105-115

УДК 519.72

Матричное представление частотного словаря для восстановления отсутствующих данных

Антон Г.Рубцов*
Мария Ю.Сенашова†
Институт вычислительного моделирования СО РАН,
Академгородок 50, Красноярск, 660036,
Россия

Проблема восстановления утерянных данных актуальна как для фундаментальных, так и для прикладных областей науки. Результаты восстановления существенно зависят от способа, которым восстановление производилось, от характера самих данных и от утерянных данных. Некоторые утерянные данные не могут быть восстановлены никакими разумными способами. Любые данные, по крайней мере теоретически, всегда можно рассматривать как символьную последовательность из конечного алфавита. В рамках настоящей работы рассматриваются лишь такие данные, что не уменьшает общности представленных результатов.

Ключевые слова: восстановление отсутствующих данных, частотный словарь, условная энтропия.

Восстановление отсутствующих данных является важной прикладной задачей в различных областях естествознания и техники [1]. Общие подходы к этой проблеме представляют собой одну из фундаментальных проблем вычислительной математики. Результаты восстановления отсутствующих данных зависят как от способа восстановления, так и от характера самих данных. Ранее были предложены подходы к решению этой проблемы, основывающиеся на идее моделирования работы высоко параллельных мелкозернистых вычислительных устройств [2, 3]. Настоящая работа продолжает этот цикл исследований; мы изложим некоторые результаты, связанные с восстановлением отсутствующих данных на основе принципа максимального подобия. Данный принцип может быть представлен как экстремальный — принцип максимума условной энтропии.

Предполагаем, что алфавит, в котором записаны изучаемые последовательности, заранее известен, а сами данные представляют собой последовательность символов. Теоретически любые данные можно свести к этой форме. Отсутствие части такой последовательности будем рассматривать как потерю данных. При этом будем считать, что длина утерянной части известна, а сама отсутствующая часть является связным диапазоном. Причем восстанавливать отсутствующую часть будем, используя только имеющиеся в наличии данные (небольшие фрагменты присутствующих частей последовательности). В такой постановке проблема весьма актуальна для самых различных областей знания — от теории передачи данных до молекулярной биологии. В данной работе изложены результаты, связанные с восстановлением таких данных на основе принципа максимального подобия.

Постановка задачи. Рассматривается символьная последовательность, состоящая из символов алфавита Ω. Пусть L — длина участка, который необходимо восстановить (будем называть его лакуной). Словом длины q будем называть любую связную последовательность этой длины, составленную из символов алфавита Ω. Опорным частотным словарем

*e-mail: Anton.Rubtsov@rusal.com
†e-mail: msen@icm.krasn.ru

W толщины q будем называть список всех слов этой длины, встречающихся в доступных исследователю частях, с указанием частот этих слов f_ω. Частота $f_\omega = \frac{n}{N}$, где N — количество всех слов данной длины в символьной последовательности, а n — количество в последовательности данного конкретного слова ($0 < f_\omega \leq 1$). Словарь W называется полным, если он содержит слова, состоящие из всех возможных сочетаний символов алфавита Ω. В противном случае словарь неполный. Пополненным частотным словарем $\overline{W}$ будем называть частотный словарь (толщины q), составленный по той последовательности, которая возникает в результате заполнения лакуны. Пополненный частотный словарь будет использоваться для выбора из всех возможных заполнений, наиболее «подходящего» для данной символьной последовательности. Критерии выбора такого заполнения могут быть различными. Наиболее естественным, с нашей точки зрения, является принцип минимума условной энтропии опорного частотного словаря относительно пополненного (принцип максимального подобия). Условная энтропия ([4]) для конкретного заполнения вычисляется следующим образом:

$$S = \sum_i f_i \ln\left(\frac{f_i}{\widetilde{f}_i}\right).$$

Здесь сумма берется по всем словам, встречающимся в полученном тексте, f_i — частота слова в опорном словаре, $\widetilde{f}_i$ — частота слова в пополненном словаре. Критерий минимума условной энтропии позволяет выбирать такое заполнение, которое максимально похоже на имеющиеся части последовательности.

Левой α_l (соответственно, правой α_r) опорой длины t, $0 \leq t \leq q-1$ будем называть слово этой длины, расположенное сразу слева (соответственно, сразу справа) от лакуны. Наиболее целесообразным представляется выбор опоры длиной $q-1$, поскольку опора такой длины позволяет максимально использовать при заполнении информацию имеющихся частей символьной последовательности. Однако если символьная последовательность не позволяет использовать опору такой длины (в частотном словаре нет слов, которые могут быть присоединены к такой опоре), можно выбирать опору меньшей длины. При этом имеет смысл выбирать опору максимальной длины из возможных.

Поскольку нас будут интересовать только те заполнения лакуны, которые можно получить с использованием опорного частотного словаря, длина восстанавливаемой символьной последовательности фактически будет составлять $L+2t$ символа, включая левую и правую опоры.

Построить заполнение лакуны означает построить последовательность символов, которая строится при помощи слов длины q

$$\omega_1, \omega_2, \omega_3, ..., \omega_{L+2t-q}, \omega_{L+2t-q+1}, \tag{1}$$

причем для каждой пары соседних слов выполняется условие:

$$\omega_j = i_1\bar{\omega}; \quad \bar{\omega}i_q = \omega_{j+1}; \tag{2}$$

т.е. два соседних слова пересекаются по общему подслову длины $q-1$, а первое слово в этой последовательности (соответственно, последнее) начинается (соответственно, заканчивается) левой опорой α_l (соответственно, правой опорой α_r). Длина такой последовательности составит $L+2t$ символа. Для построения последовательности (1) на первом шаге к какой-либо из опор (возьмем для определенности левую опору α_l) присоединим слово из частотного словаря, у которого первые $q-1$ символов совпадают с опорой. Присоединение слова

состоит в приписывании к присоединяемому слову последнего символа присоединяемого слова. В общем случае слов из словаря, которых можно присоединить к опоре, найдется несколько. Нас интересуют все возможные заполнения лакуны, поэтому мы должны присоединить все такие слова. Следующим шагом к каждому полученному продолжению опоры присоединяем те слова из словаря, у которых первые $q-1$ символов совпадают с последними $q-1$ символами полученных цепочек. Процесс продолжается до тех пор, пока длина полученных цепочек символов не станет равна $L+2t$, поскольку нас интересуют заполнения, опирающиеся на известные части последовательности.

Процесс построения заполнений можно представить в виде роста дерева. В корне этого дерева находится слово, совпадающее с левой опорой. Каждый узел — это слово длины q, которое получается присоединением одного символа к последним $q-1$ символам слова родительского узла. Из каждого узла выходит столько ветвей, сколько можно присоединить слов из опорного словаря к текущему слову. Глубина такого дерева будет составлять $L+t$. Так как при построении заполнений часто возникают ситуации, когда на некотором шаге к слову нельзя присоединить ни один символ, то, соответственно, дерево может быть неполным. Введем определения, касающиеся дерева заполнений.

Определение 1. *Дерево, включающее в себя как ветки, которые могут дорасти до глубины $L+t$, так и ветки, которые обрываются на некотором уровне, будем называть полным деревом.*

Определение 2. *Дерево, включающее в себя только те ветки, которые достигают глубины $L+t$, будем называть усеченным деревом.*

Чтобы по построенному дереву получить интересующие нас заполнения, нужно просмотреть все листья на глубине $L+t$ и выбрать те из них, у которых последние $q-1$ символов слова совпадают с правой опорой α_r. Число всех возможных заполнений для полного словаря W составляет $\aleph^L$, где $\aleph$ — мощность алфавита Ω. Это число служит верхней границей количества возможных заполнений. Для неполного словаря число возможных заполнений меньше. Задача построения всех возможных заполнений является задачей полного перебора, а следовательно, очень ресурсоемкой, поэтому нужен способ построения только тех заполнений, у которых правые $q-1$ символов совпадают с правой опорой α_r, минуя построение всего дерева заполнений.

Построение заполнений с помощью матричного представления частотного словаря. Способ построения только интересующих нас заполнений дает специальное представление частотного словаря. Всякий частотный словарь есть (упорядоченный) список слов длины q. Такой список можно однозначно преобразовать в матрицу $\mathfrak{A}$ порядка $\aleph^q \times \aleph^q$, в которой вспомогательная строка и столбец помечены словами опорного словаря. Если последние $(q-1)$ символов слова ω_i из i-й строки совпадают с первыми $(q-1)$ символами слова ω_j из j-го столбца, то на пересечении i-й строки и j-го столбца записывается последний символ слова ω_j. Вспомогательные строка и столбец в вычислениях не участвуют.

Полученную матрицу будем называть матрицей заполнений (табл. 1). Для построения заполнений возведем данную матрицу в степень $L+t+1$. Так как элементами матрицы являются символы, то для того, чтобы возвести матрицу данного вида в степень, переопределим операции сложения и умножения для элементов матрицы. Под сложением будем понимать конкатенацию строк при помощи служебного символа, не входящего в алфавит Ω. Суммой

Таблица 1. Матрица заполнений

	$\nu_1\overline{\omega_1}$	...	$\overline{\omega_1}\nu_1$	$\overline{\omega_1}\nu_2$	$\overline{\omega_1}\nu_3$	...	$\overline{\varphi_1}\nu_1$	$\overline{\varphi_1}\nu_2$	$\overline{\varphi_1}\nu_3$	...
$\nu_1\overline{\omega_1}$	”	...	ν_1	ν_2	ν_3	...	”	”	”	...
$\nu_1\overline{\omega_2}$	”	...	”	”	”	...	”	”	”	...
$\nu_1\overline{\omega_3}$	”	...	”	”	”	...	”	”	”	...
...	...	...	...	...	...	...	...	...	...	...
$\nu_i\overline{\varphi_1}$	”	...	”	”	”	...	ν_1	ν_2	ν_3	...
$\nu_i\overline{\varphi_2}$	”	...	”	”	”	...	”	”	”	...
$\nu_i\overline{\varphi_3}$	”	...	”	”	”	...	”	”	”	...

двух строк "00000"и "11111" будет строка следующего вида "00000+11111 где ‘+’ — служебный символ. Произведением двух строк будет строка, полученная по следующему правилу: к каждой строке первого множителя, находящейся между служебными символами, приписывается каждая строка второго множителя, находящаяся между служебными символами. Например, первый множитель — строка "001+000" , второй — строка "011+100". Тогда произведением этих строк является строка "001011+000011+001100+000100".

Теорема 1. *Умножение матрицы заполнений на себя эквивалентно росту множества усеченных деревьев. Причем степень, в которую возводится матрица, плюс один совпадает с глубиной деревьев, то есть с каждым умножением матрицы на себя глубина деревьев увеличивается на единицу.*

Доказательство. Для доказательства теоремы построим процедуру, с помощью которой можно будет из матрицы некоторой степени n получить дерево соответствующей глубины и наоборот.

Пусть имеется некоторый текст T, составленный из конечного алфавита символов $\Omega = \{\nu_1, \nu_2 \dots \nu_k\}$, и соответствующий ему частотный словарь $W(q)$ некоторой толщины q. Будем полагать, что данный словарь является полным, то есть он содержит все возможные слова длины q, которые можно построить из символов Ω. Словарь $W(q)$ можно рассматривать как набор $W(q) = \{\bigcup_{j=1}^{q} \nu_{k_{ij}}\}$, где $\nu_{k_{i1}}, \nu_{k_{i2}}, ..., \nu_{k_{il}}$ — символы алфавита Ω, $i1, i2, ..., il$ — номера символов в алфавите Ω, $k_{i1}, k_{i2}, ..., k_{il}$ — некоторая перестановка индексов, k — номер символа в слове. Каждое слово словаря $W(q)$ можно представить в следующем виде:

$$\bigcup_{j=1}^{q} \nu_{k_{ij}} = \omega_{lj}\nu_{q_{ij}} = \nu_{1_{ij}}\omega_{rj},$$

здесь ω_{lj} — множество левых частей слов длиной $q-1$, а ω_{rj} — множество правых частей. Пусть в тексте T отсутствует часть длиной L. Обозначим длину левой опоры через $left = q-1$, а сама опора есть некоторое слово $\bigcup_{j=1}^{q-1} \nu_{k_{im}}$, обозначим его через $\overline{\omega}$. Тогда дерево заполнений будет выглядеть, как показано на рис. 1.

Слово $\overline{\omega}$ (левая опора) обязательно будет являться правой частью некоторого множества слов (как минимум, такое слово будет одно) словаря $W(q)$, так как словарь $W(q)$ построен по всему тексту T, а $\overline{\omega}$ — часть этого текста. Поэтому слова словаря $\nu_{1_{i1}}\omega_{r1}, \nu_{1_{i2}}\omega_{r2}, \dots, \nu_{1_{il}}\omega_{r|W|}$ могут рассматриваться как всевозможные варианты левых опор в данном тексте T.

Для определенности и без потери общности возьмем конкретный вид левой части (конкретное слово из опорного частотного словаря) и построим дерево заполнений.

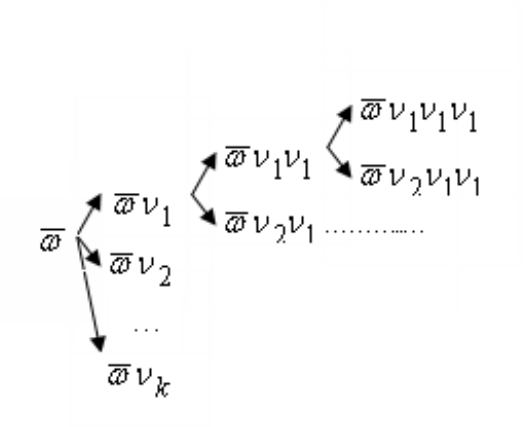

Рис. 1. Дерево заполнений

Пусть $\nu\varphi_r$ — некоторое слово из словаря $W(q)$, причем φ_r — подслово длины $q-1$, совпадающее с левой опорой. Так как словарь полный, то на первом шаге к данной опоре может быть присоединено не более чем $|W|$ слов. К тому же часть слов может не иметь той общей части, по которой опора и слово могли бы пересекаться. Пусть подобных слов на первом шаге s_1. Таким образом, на первом шаге к опоре $\nu\varphi_r$ может присоединиться $|W|-s_1$ слов словаря W. Обозначим множество этих слов $\{\varphi_r\nu_i\}$, где φ_r — общая часть слова, по которой осуществляется пересечение с опорой, а ν_i — некоторый символ алфавита Ω. Рост дерева заполнений показан на рис. 2.

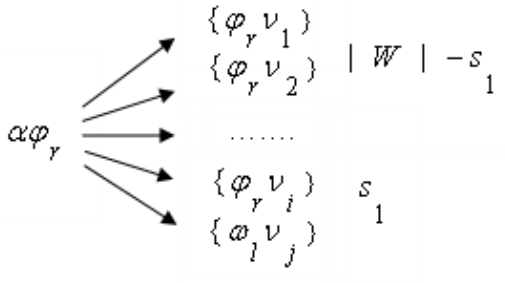

Рис. 2. Схема роста дерева заполнений

Представим полученное дерево в виде табл. 2. Слово $\nu\varphi_r$, содержащее левую опору, разместим во второй строке и первом столбце. В первой строке по столбцам расположим слова словаря W. Если слово $\nu\varphi_r$ имеет общую часть (длиной $q-1$) со словом из j -го столбца, то на пересечении этого столбца со строкой ставим последний символ слова из j -го столбца. Если опора не имеет общей части с каким-либо словом, то в соответствующей

Таблица 2. Представление дерева

	$\omega_l\nu_j$	...	$\varphi_r\nu_1$	$\varphi_r\nu_2$	...	$\varphi_r\nu_{\lvert W\rvert-s1}$	$\omega_l\nu_j$
$\nu\varphi_r$		...	ν_1	ν_2	...	$\nu_{\lvert W\rvert-s1}$	

ячейке будет пустой символ. Понятно, что таких столбцов будет $|W|$.

Так как словарь полный и отсортирован по алфавиту, то слова $\varphi_r\nu_1$, $\varphi_r\nu_2, \ldots, \varphi_r\nu_{|W|-s1}$ будут расположены сразу друг за другом, но необязательно начиная с первого столбца. В итоге будет получена таблица, которая соответствует дереву заполнений на первом шаге и однозначно его определяет. При этом глубина дерева равна единице.

Продолжим построение дерева дальше. На следующем шаге существует $|W|-s_1$ продолжений вида $\varphi_r\nu_1, \varphi_r\nu_2, \ldots, \varphi_r\nu_{|W|-s1}$, к которым могут быть присоединены слова словаря $W(q)$. Так как присоединение символов осуществляется справа, то представим множество продолжений $\{\varphi_r\nu_i\}$ как $\{\nu_i'\varphi_r'\}$. Так же формально учитываем ветки, которые не имели продолжения. Это касается тех слов, которые не присоединились, т.е. $\{\omega_l\nu_j\} = \{\nu_j'\omega_l'\}$.

К каждому заполнению, полученному на первом шаге, последовательно подбираем все слова словаря и отмечаем те, которые могут присоединиться:

$$\{\nu_i'\varphi_r'\}\bigcup\{\nu_j'\omega_l'\} = \{\nu_1\varphi_1, \nu_2\varphi_2, \ldots, \nu_{|W|-s1}\varphi_{|W|-s1}, \ldots, \nu_1\omega_1, \ldots, \nu_{s1}\omega_{s1}\},$$

причем $\nu_1\omega_1$,...,$\nu_{s1}\omega_{s1}$ учитываются формально.

Напомним операции умножения и сложения слов. Если слово $\nu_i\varphi_k$ имеет общую часть справа длиной $q-1$ с другим словом $\varphi_k\nu_j$, то результатом умножения будет слово вида $\nu_i\varphi_k * \varphi_k\nu_j = \nu_i\varphi_k\nu_j$. Если общей части нет, то результат есть пустое множество (пустая строка). Результат сложения двух строк есть третья строка "$\nu_1\varphi_1$"+ "$\varphi_1\nu_1$"= "$\nu_1\varphi_1+\varphi_1\nu_1$". Прибавление пустой строки исходной строки не меняет: "$\nu_1\varphi_1$"+ ""="$\nu_1\varphi_1$". Таким образом, получим последовательность операций, изображенных на рис. 3, 4.

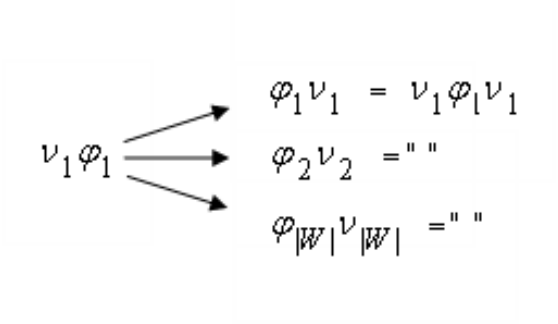

Рис. 3. Подбор слов к первому заполнению

Пустой символ означает, что слово из словаря не может быть присоединено к данному заполнению и ветка дерева прекращает рост.

Все эти потенциальные продолжения разобьем на классы или группы. Причем присоединение будем обозначать знаком умножения, а объединение — знаком сложения.

В первую группу $G_1(\varphi_1\nu_1)$ попадут возможные продолжения с помощью слова $\varphi_1\nu_1$:

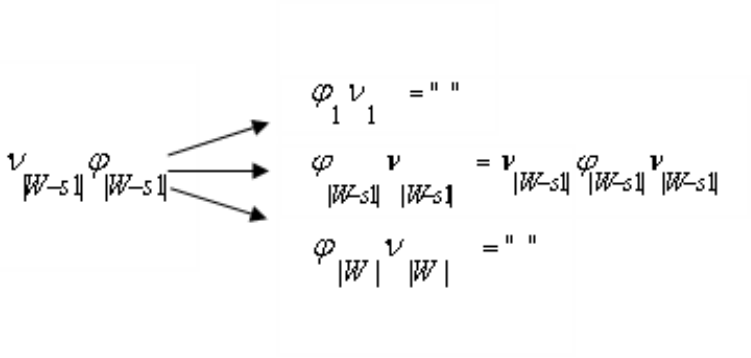

Рис. 4. Подбор слов к заполнению $(W - s_1)$

$\nu_1\varphi_1 * \varphi_1\nu_1 = \nu_1\varphi_1\nu_1;$
$\nu_2\varphi_2 * \varphi_1\nu_1 =$ “;
. . . . . ;
$\nu_{|W|-s_1}\varphi_{|W|-s_1} * \varphi_1\nu_1 =$ “;

Следующих веток фактически нет, но формально мы их учитываем.

$\nu_1\omega_1 * \varphi_1\nu_1 =$ “;
. . . . . ;
$\nu_{s_1}\omega_{s_1} * \varphi_1\nu_1 =$ “;

$G_1(\varphi_1\nu_1) = \{\nu_1\varphi_1 * \varphi_1\nu_1 + \nu_2\varphi_2 * \varphi_1\nu_1 + \cdots + \nu_{|W|-s_1}\varphi_{|W|-s_1} * \varphi_1\nu_1 + \nu_1\omega_1 * \varphi_1\nu_1 + \nu_2\omega_2 * \varphi_1\nu_1 + \cdots + \nu_{s_1}\omega_{s_1} * \varphi_1\nu_1\}.$

Во вторую группу $G_2(\varphi_2\nu_2)$ попадут возможные заполнения с помощью слова $\varphi_2\nu_2$:

$\nu_1\varphi_1 * \varphi_2\nu_2 =$ “;
$\nu_2\varphi_2 * \varphi_2\nu_2 = \nu_2\varphi_2\nu_2;$
. . . . . ;
$\nu_{|W|-s_1}\varphi_{|W|-s_1} * \varphi_2\nu_2 =$ “;

Нижеследующих веток фактически нет, но формально мы их учитываем.

$\nu_1\omega_1 * \varphi_2\nu_2 =$ “;
. . . . . ;
$\nu_{s_1}\omega_{s_1} * \varphi_2\nu_2 =$ “”;

$G_2(\varphi_2\nu_2) = \{\nu_1\varphi_1 * \varphi_2\nu_2 + \nu_2\varphi_2 * \varphi_2\nu_2 + \cdots + \nu_{|W|-s_1}\varphi_{|W|-s_1} * \varphi_2\nu_2 + \nu_1\omega_1 * \varphi_2\nu_2 + \nu_2\omega_2 * \varphi_2\nu_2 + \cdots + \nu_{s_1}\omega_{s_1} * \varphi_2\nu_2\}$
И так далее...

$G_{|W|-s_1}(\varphi_{|W|-s_1}\nu_{|W|-s_1}) = \{\nu_1\varphi_1 * \varphi_{|W|-s_1}\nu_{|W|-s_1} + \nu_2\varphi_2 * \varphi_{|W|-s_1}\nu_{|W|-s_1} + \cdots + \nu_{|W|-s_1}\varphi_{|W|-s_1} * \varphi_{|W|-s_1}\nu_{|W|-s_1} + \nu_1\omega_1 * \varphi_{|W|-s_1}\nu_{|W|-s_1} + \nu_2\omega_2 * \varphi_{|W|-s_1}\nu_{|W|-s_1} + \cdots + \nu_{s_1}\omega_{s_1} * \varphi_{|W|-s_1}\nu_{|W|-s_1}\}$

. . . . . .

$G_{|W|}(\varphi_{|W|}\nu_{|W|}) = \{\nu_1\varphi_1 * \varphi_{|W|}\nu_{|W|} + \nu_2\varphi_2 * \varphi_{|W|}\nu_{|W|} + \cdots + \nu_{|W|}\varphi_{|W|} * \varphi_{|W|}\nu_{|W|} + \nu_1\omega_1 * \varphi_{|W|}\nu_{|W|} + \nu_2\omega_2 * \varphi_{|W|}\nu_{|W|} + \cdots + \nu_{s_1}\omega_{s_1} * \varphi_{|W|}\nu_{|W|}\}.$

Таблица 3. Определение продолжений

	$\omega_l\nu_j$	...	$\varphi_r\nu_1$	$\varphi_r\nu_2$	...	$\varphi_r\nu_{\lvert W\rvert-s1}$	$\omega_l\nu_j$
$\nu\varphi_r$		...	ν_1	ν_2	...	$\nu_{\lvert W\rvert-s1}$	

Таблица 4. Результат умножения

	$\varphi_1\nu_1$
$\nu_1\varphi_1$	ν_1
$\nu_2\varphi_2$	
$\nu_3\varphi_3$	
...	...
$\nu_{\lvert W\rvert-s_1}\varphi_{\lvert W\rvert-s_1}$	
$\nu_1\omega_1$	
...	...
$\nu_j\omega_l$	

Каждую из полученных сумм представим в виде произведения двух векторов G_1:

$$G_1(\varphi_1\nu_1) = \{\nu_1\varphi_1 * \varphi_1\nu_1 + \nu_2\varphi_2 * \varphi_1\nu_1 + \cdots + \nu_{|W|-s_1}\varphi_{|W|-s_1} * \varphi_1\nu_1 +$$
$$+\nu_1\omega_1 * \varphi_1\nu_1 + \nu_2\omega_2 * \varphi_1\nu_1 + \cdots + \nu_{s_1}\omega_{s_1} * \varphi_1\nu_1\} =$$

$$= (\nu_1\varphi_1, \nu_2\varphi_2, \ldots, \nu_{|W|-s_1}\varphi_{|W|-s_1}, \nu_1\omega_1, \ldots, \nu_{s_1}\omega_{s_1}) \times \begin{pmatrix} \varphi_1\nu_1 \\ \varphi_1\nu_1 \\ \cdots \\ \varphi_1\nu_1 \end{pmatrix} = \begin{pmatrix} \nu_1\varphi_1\nu_1 \\ \\ \cdots \\ \\ \end{pmatrix} \quad (3)$$

Продолжения, полученные на первом шаге, а соответственно и вектор слева в выражении 3 определяются табл. 3.

Вектор справа и итоговый вектор можно записать в виде табл. 4, где в столбце справа расположены символы, которые могут быть присоединены с помощью слова $\varphi_1\nu_1$, а в столбце слева располагаются слова, к которым можно присоединить соответствующий символ. Аналогично выглядят таблицы для всех других присоединяемых слов $\varphi_2\nu_2, \varphi_3\nu_3, \ldots, \varphi_{|W|}\nu_{|W|}$. Объединив все такие таблицы, получим табл. 5.

Как уже говорилось, каждый столбец данной таблицы определяет множество символов, которые могут быть присоединены к определенному слову, находящемуся в строке, с помощью слова по столбцу. Понятно, что непустых ячеек в таблице в каждой строке может быть и больше одной.

Матрицу, которую образует данная таблица будем называть матрицей продолжений (табл. 6). Она определяет какие символы, могут быть присоединены к продолжениям и с помощью каких слов.

Таким образом, множество сумм $G_1, G_2, \ldots, G_{|W|}$ получается путем умножения вектора заполнений, найденного на первом шаге, на матрицу продолжений (табл. 7). Причем порядки следования слов друг за другом по столбцам и строкам должны совпадать.

Таблица 5. Таблица заполнений на первом шаге

	$\varphi_1\nu_1$	$\varphi_2\nu_2$	$\varphi_3\nu_3$	$\dots$	$\varphi_{\lvert W\rvert-s_1}\nu_{\lvert W\rvert-s_1}$
$\nu_1\varphi_1$	ν_1				
$\nu_2\varphi_2$		ν_2			
$\nu_3\varphi_3$			ν_3		
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
$\nu_{\lvert W\rvert-s_1}\varphi_{\lvert W\rvert-s_1}$					$\nu_{\lvert W\rvert-s_1}$
$\nu_1\omega_1$					
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
$\nu_j\omega_l$					

Таблица 6. Матрица заполнений на первом шаге

ν_1				
	ν_2			
		ν_3		
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$
				$\nu_{\lvert W\rvert-s_1}$
$\dots$	$\dots$	$\dots$	$\dots$	$\dots$

Таблица 7. Вектор заполнений на первом шаге

	$\dots$	ν_1	ν_2	$\dots$	$\nu_{\lvert W\rvert-s_1}$	$\dots$	

В результате умножения получим табл. 8.

Таблица 8. Матрица заполнений

	$\omega_l\nu_j$	$\dots$	$\varphi_r\nu_1$	$\varphi_r\nu_2$	$\dots$	$\varphi_r\nu_{\lvert W\rvert-s_1}$	$\dots$	$\omega_l\nu_j$
$\alpha\varphi_r$		$\dots$	G_1	G_2	$\dots$	$G_{\lvert W\rvert-s_1}$	$\dots$	

Данная таблица определяет множество продолжений, которые образовались на втором шаге. Каждая непустая клетка содержит сумму (объединение) слов, которые получились присоединением слова, расположенного в верхней строке соответствующего столбца. При этом длина заполнений увеличилась еще на единицу и стала равной двум.

Все заполнения, принадлежащие одной клетке, имеют одинаковые окончания, так как были получены в результате присоединения одного общего для них слова.

Умножая вектор

$$\{\dots G_1, G_2, \dots, G_{|W|-s_1}, \dots\}$$

на матрицу продолжений, в соответствии с правилами умножения и сложения строк, получим следующий вектор, содержащий все заполнения на 3-м шаге, и так далее.

Проделав $L+t+1$ таких умножений, получим матрицу, содержащую все заполнения этой же длины, которые только возможно получить по данному частотному словарю из опоры $\nu\varphi_r$. При этом заполнения, которые имеют общую часть длиной $q-1$ с правой опорой, содержатся в столбце, в котором левые $q-1$ символов слова совпадают с правой опорой.□

Результаты. Изучение качества восстановления проводилось с помощью вычислительных экспериментов по заполнению лакун в символьной последовательности. В качестве тест-объектов использовались следующие символьные последовательности:

1) двоичные последовательности;
2) генетические последовательности.

Результаты получены, исходя из возможностей персонального компьютера.

Четырехбуквенные последовательности. В качестве рассматриваемой последовательности был взят текст генетической последовательности (AB012132 — геном вируса парагриппа человека). Длина текста составляла 14573 символа, длина заполнения — 7 символов. Толщина словаря — 3 символа. Часть строки интересующей нас ячейки матрицы, полученной в результате возведения в седьмую степень, приведена ниже:

aaaaagta + caaaagta + gaaaagta + taaaagta + acaaagta + ccaaagta + gcaaagta + tcaaagta + agaaagta + cgaaagta + ggaaagta + tgaaagta + ataaagta + ctaaagta + gtaaagta + ttaaagta + aacaagta + cacaagta + gacaagta + tacaagta + accaagta + cccaagta + gccaagta + tccaagta + agcaagta + cgcaagta + ggcaagta + tgcaagta + atcaagta + ctcaagta + gtcaagta + ttcaagta + aagaagta + cagaagta + gagaagta + tagaagta + acgaagta + ccgaagta + gcgaagta + tcgaagta + aggaagta + cggaagta + gggaagta + tggaagta + atgaagta + ctgaagta + gtgaagta + ttgaagta + aataagta + cataagta + gataagta + tataagta + ...

Заполнение	Абс. энтропия	Условная энтропия
gtgcagt - точное	3,931363626427	9,086736165523E-07
ggcgcgt	Макс. 3,931962720496	8,895312249570E-06
gaaatgt	3,931068694792	Мин. 3,604751137360E-07

Двухбуквенные последовательности. Последовательность из алфавита {0, 1} получалась выписыванием подряд без пробелов натуральных чисел, записанных в двоичной системе счисления: 1101110010111011, 11000100110101011... и так далее. Длина лакуны составляла 11 символов. Часть значения строки нужной нам ячейки матрицы, полученной в результате возведения в нужную степень, приведена ниже:

0000000000110 + 1000000000110 + 0100000000110 + 1100000000110 + 0010000000110 + 1010000000110 + 0110000000110 + 1110000000110 + 0001000000110 + 1001000000110 + 0101000000110 + 1101000000110 + 0011000000110 + 1011000000110 + 0111000000110 + 1111000000110 + 0000100000110 + 1000100000110 + 0100100000110 + 1100100000110 + 0010100000110 + 1010100000110 + 0110100000110 + 1110100000110 + 0001100000110 + 1001100000110 + 0101100000110 + 1101100000110 + 0011100000110 + 1011100000110 + 0111100000110 + 1111100000110 + 0000010000110 + 1000010000110 + ...

Заполнение	Абс. энтропия	Условная энтропия
0011001100110 - точное	2,0722135611	1,4701216237E-06
0000000000110	Макс. 2,0726107030	8,2417286609E-06
0111101000110	2,0721937671	Мин. 1,2353260849E-7

Выводы. В настоящей работе получен принципиально новый метод построения заполнений в символьных последовательностях. Матричное представление частотного словаря позволяет получить все возможные для данной лакуны и данного словаря заполнения.

Работа выполнена при финансовой поддержке гранта Президента РФ для ведущих научных школ №НШ-3431.2008.9.

Список литературы

[1] A.N.Gorban, D.A.Rossiev, D.C.Wunsch II, Neural Network Modelling of Data with Gaps, *Радиоэлектроника. Информатика. Управление,* (2000), №1, 47-55.

[2] М.Ю.Сенашова, М.Г.Садовский, А.Г.Рубцов, Кинетическая машина Кирдина в проблеме восстановления отсутствующих фрагментов символьных последовательностей, *Ползуновский альманах*, Барнаул, (2006), №11, 131-133.

[3] М.Ю.Сенашова, А.Г.Рубцов, М.Г.Садовский, Применение кинетической машины Кирдина для восстановления утерянных данных в символьных последовательностях, *Информационные и математические технологии в научных исследованиях*, Труды XI международной конференции "Информационные и математические технологии в научных исследованиях", 2006, Иркутск, ИСЭМ СО РАН, Часть II, 168-176.

[4] А.Н.Горбань, Обход равновесия, Наука, Новосибирск, 1984.

Data Loss Recovery at Symbolic Sequences Due to Matrix Formulation of Frequency Dictionary

Anton G.Rubtsov
Mariya Yu.Senashova

Data loss recovery is provided due to the principle of maximal likelihood. The matrix representations of the frequency dictionary allow one to calculate the number of possible fillings and their probabilities. The method is applied to various areas of knowledge, for example to communication theory and molecular biology.

Keywords: data loss recovery, frequency dictionary, relative entropy